SFA Modernization Partner

United States Department of Education

Student Financial Assistance

Integrated Technical Architecture

Detailed Design Document

Executive Summary

Task Order #16

Deliverable # 16.1.2

October 13, 2000

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1. Purpose

The Integrated Technical Architecture (ITA) was developed in support of the Department of Education's (DOE) Modernization Program. The ITA provides a rich set of services providing standard hardware and software product platforms and functionality to access existing Student Financial Assistance (SFA) stovepipe mainframe legacy systems through web-enabled applications. This standardization of the architecture provides reduced cost of ownership through common services and features, reusability of common infrastructure and application components, and the definition of object based development and operational environments.

1.2. Scope

The ITA is a strategic component of the overall SFA enterprise architecture. It is comprised of four core architecture domains: Internet, Integration, Data Warehouse and Security. The initial Release 1.0 of the architecture will evolve as existing systems are modified or retired and new packaged or Internet-based applications are added to the SFA environment.

In the implementation of the ITA, three major environments (execution, development, and operations) must be integrated in order to support future business capabilities.

The **Execution Architecture** is comprised of run-time services required when an application executes.

The **Operations Architecture** is a combination of tools, support services, procedures, and controls required to keep a production system up and running efficiently.

The **Development Architecture** is the environment for one or several systems development projects as well as for maintenance efforts.

The Integrated Technical Architecture design document provides the definition and services for each of the three environments – execution, operations and development.

1.3. Document Organization

This ITA Detailed Design Document contains this executive summary and seven (7) subsequent volumes. The detailed design document is comprised of the following volumes,

- Executive Summary
- Volume 1 – ITA Conceptual Architecture
- Volume 2 – ITA Internet Architecture

- Volume 3 – ITA Enterprise Architecture Implementation (EAI) Architecture
- Volume 4 – ITA Data Warehouse Architecture
- Volume 5 – ITA Security Architecture
- Volume 6 – ITA Development Architecture
- Volume 7 – ITA Operations Architecture

The Executive Summary is contained within this section and provides an overview and a high level discussion of the Integrated Technical Architecture and an overview of each volume of the design document.

Volume 1, the ITA Conceptual Architecture defines the Conceptual Architecture on which the ITA was designed and developed.

The Execution Architecture defines the components required for the applications running in the ITA to execute. The Execution Architecture of the ITA is divided into 4 distinct sections, Internet, EAI, Data Warehouse, and Security. Volumes 3 – 6 of the ITA Detailed Design document define the execution architecture components.

Volume 2, the ITA Internet Architecture includes the components, services and the functionality of these components to define the architecture and integration for supporting SFA applications utilizing these services.

Volume 3, the ITA EAI Architecture defines the set of technology services that enables the sharing of processes and data of disparate systems to support end-to-end business processes. The EAI Architecture enables the many “stovepipe” applications to exchange information via common, reusable methods and infrastructure. The EAI will allow SFA to integrate new net-centric applications with existing back-end systems, while at the same time, providing a means to migrate away from reliance upon these legacy systems.

Volume 4, the ITA Data Warehouse includes the components, services and the functionality of these components to define the architecture and integration for supporting SFA Data Warehousing applications.

Volume 5, the ITA Security Architecture defines the framework from which an enterprise security architecture to provide the necessary security services and functionality to adequately protect the ITA applications and data.

Volume 6, the ITA Development Environment defines the development tools, methods, standards, and procedures that define the development environment for the ITA. The purpose of the development architecture is to support the tasks involved in the analysis, design, construction, and maintenance of SFA business applications. This volume will address the development tools and architecture required to support the development of applications. The standards, methods and procedures for development are not included in this volume.

Volume 7, the ITA Operations Architecture defines the components required to operate and manage the ITA once deployed to production. The operations architecture is a set of tools, support services procedures and controls required to keep the production system up and running efficiently.

2 Integrated Technical Architecture

The objective of the Integrated Technical Architecture is to Design, Build & Deploy the technical services, infrastructure, and components required to enable the development of business applications.

The Office of Student Financial Assistance is moving towards a service-oriented architecture that will enable its modernized business capabilities. SFA has developed a technical architecture framework and identified the technology services required to enable migration to this new, modernized environment. The ITA will be the technical foundation that will support all of the Modernization Program's re-engineered business process and systems improvements. The initial Release 1.0 of the ITA will result in the following outcomes:

- Technical architecture detailed designs (development, execution, and operations environments) for the Internet, Integration, Data Warehouse and Security architectures

- Installed and configured technical architecture products and services for the Internet, Integration, Data Warehouse and Security architectures

- Tested, stable integrated technical architecture and core services (based on the Internet, Integration, Data Warehouse and Security architectures / services)

Release 1.0 of the ITA is the building block for future phases of the technical architecture. Once the core technology components are in place, SFA business capabilities will be able to begin building on top of the ITA.

The ITA will provide a standardized, reusable infrastructure for enabling business capabilities. Future releases will build upon Release 1.0, and new technology capabilities will be added, as required. The long-term vision is to provide an integrated, enterprise-wide technical architecture that will enable SFA to reduce the number of custom-built, siloed applications that are difficult to update and maintain. The technical architecture will provide the foundation for SFA to move towards a Capability Maturity Model (CMM) Level 3 rating.

Figure 1 below depicts the ITA context diagram.

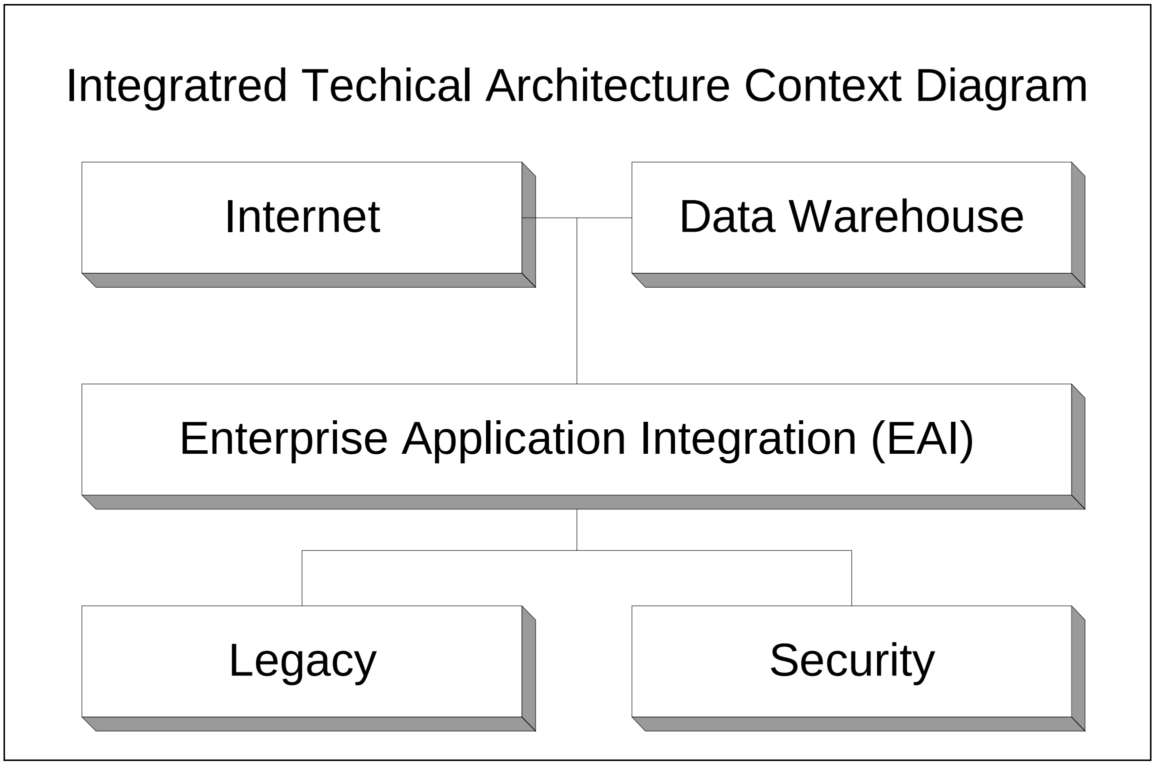

Integratred Techical Architecture Context Diagram
Internet
Data Warehouse
Enterprise Application Integration (EAI)
Legacy
Security

Figure 1 –Integrated Technical Architecture Context

The following sections of this document provide a high level overview of each of the ITA Detailed Design Document volumes and their content.

3 Conceptual Architecture

The purpose of the Conceptual Architecture document is to define the architectural foundation on which future applications at Department of Education are to be developed. This document is intended to provide Department of Education's application architects with a blue print for developing systems which meet the company's long term computing goals. It is not the purpose of this document to provide an exhaustive inventory SFA existing application portfolio, nor is the intent of this document to address in a complete manner all aspects of the application development life cycle. Each element of the architecture described within this volume requires logical predecessor / follow-on activities to ensure their realisation, adoption and risk management.

3.1. Business Imperatives

The following business imperatives resulted in the necessity for the Department of Education SFA to move to an alternative technical architecture for enterprise computing.

- Technological changes that resulted in competitive pressures specifically the Internet.
- An aging IT infrastructure that is not capable of keeping pace with changing business requirements.
- SFA needed to more closely align their IT model with the business model.

By mirroring the functional layers of the Business Model the IT Model can be closely aligned and a successful technical architecture can be designed. The correlation between the layers of the Business Model and the IT Model is depicted in Figure 2.

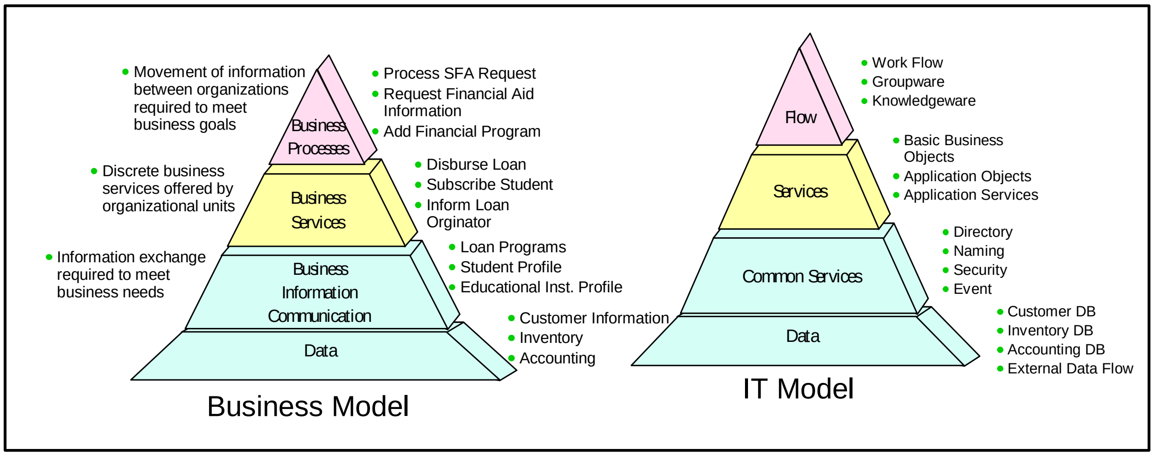


Figure 2 - Correlation between Business and IT Model Layers

3.2. Integrated Technical Architecture Strategy

The Department of Education's Integrated Technical Architecture (ITA) is driven by a need to create an environment that facilitates the integration of disparate data resources, business services and process automation thereby eliminating the barriers created by the following:

- Heterogeneous computing architectures
- Heterogeneous hardware platforms
- Heterogeneous operating systems
- Heterogeneous network configurations, topologies and protocols
- Heterogeneous representations of the same data within the enterprise

In order to meet these goals the architectural implementation must be flexible, easily modified and provide for reuse as the following changes to the enterprise computing environment are introduced:

- A shift in business processes
- Additional data resources
- Future technology shifts
- A move to provide self-service access to SFA

3.3. Design Points

The following requirements and qualifying factors influenced the design of Department of Education's SFA Integrated Technical Architecture.

3.3.1. Legacy Reuse

The Department of Education's current architecture consists of well-established legacy applications and services that are based on a centralized 2 tiered architecture. The future architecture must be able to leverage and extend this investment while enabling a flexible approach to implementing the proposed architecture. While a move to implement the proposed architecture in a short period of time is desirable, reality dictates that the current computing infrastructure will remain until the proposed architecture can be designed, prototyped and successfully implemented. Indeed some elements of the current infrastructure may remain after implementing an interface to extend services into the proposed architecture.

In order to facilitate business process modeling and data independence, additional data repositories will be implemented over time. The additional repositories will include Data

Warehouses, Data Marts and Operational Data Stores (ODS). For simplicity these new data stores will be referred to as the Enterprise Data Domain (EDD). The EDD will support a more normalized enterprise data schema than the Legacy Data Domain (LDD). Once implemented, and supported by the proper services the official instances of data will be those stored in the EDD. While current infrastructure components may remain until deprecated, data integrity must be maintained between stores for all data duplicated in both repositories (LDD and EDD).

3.3.2. Distributed Component Architecture

A Component is a piece of software that extends a known interface and provides a set of services. Services can only be provided to the client if the component interface requirements are properly satisfied. A distributed component is a service or application that is based on one or more components that are geographically distributed. The term geographically distributed refers to the architectural topology not necessarily the physical location. Distributed components could reside on the same system but reside in different regions. Distributed components must have the ability to intercommunicate in a physically and geographically distributed environment.

Distributed Components allow processing to be shared among multiple computers (or logical segments) with each computer in the network handling that portion of the overall work for which it is best suited. Distributed Components provide an excellent boundary and segmentation for implementing WorkLoad Management (WLM). Distributed Components will allow Department of Education to scale services using WLM and Load Balancing.

3.3.3. Layered Architecture

A layered architecture helps to provide structure to applications and services that can be decomposed into groups of subtasks in which each group of subtasks is at a particular level of abstraction. A layered architecture based on ever increasing abstractions of the previous layers has the following characteristics:

- Increased reusability between layers
- Decrease in ripple effect changes
- Easily modified
- Support for standardization

3.3.4. Platform Independence

With the advent of client-server computing and the Internet came the need to support different hardware and OS platforms when considering a technical architecture to support the enterprise. The predominant platforms currently used by Department of Education are as follows:

- OS 390
- VAX / VMS
- UNIX – Solaris, HP / UX
- MS-Windows, NT / 2000 / 95 / 98

The target architecture must be able to support a single development and programming environment. Applications must be deployable on each platform with a minimum of change (if any).

3.3.5. Core Support for Open/Industry Standards

Basing the future technical architecture on Open Industry Standards increases Department of Education’s ability to utilize and interact with third party products. In Addition, utilizing Open Industry Standards keeps The Department of Education from being locked into any one vendor’s solution. As part of Task Order (TO) 4, the SFA Partners selected a set of products that support DOE requirements and support open industry standards. This document defines the technical architecture mandated by the products selected in Task Order 4. A list of the standard products may be found in the TO-4 deliverables.

3.3.6. Separation of Responsibilities

The distributed component model drives decoupling the client interface framework from the business logic and data. This decoupling helps to provide a separation of responsibilities between the development staff. This allows client developers to concentrate on client frameworks and presentation logic while business logic developers work on application components. The following is a list of possible staff positions resulting from the implementation of this architecture:

- Client Developer
- Business Object Developer
- Data Object Developer or Deployment Developer
- Common Services Developer
- Database Developer

3.4. A Component-Based Technical Architecture

Department of Education’s future Technical Architecture is based on using components as building blocks to develop applications and services. Enterprise Java Bean (EJB) will be the primary standard component model used within SFA. The underlying services used by the Enterprise Java Server (EJS) and all EJBs will be supplied by Common Object Request Broker

Architecture (CORBA) services. In some cases SFA developers may supplement the EJB components with CORBA Business Objects. The technique of using components as building blocks to implement services is often referred to as a 'component framework' or in the Object-Oriented (OO) world an 'application framework.' Essentially, a component framework extends a set of interfaces and provides a well known set of services which are dictated by roles of interaction that govern how components 'plugged into' the framework may interact. The components that arise from this definition have the following characteristics:

- Are independent units of deployment
- Are units of third-party composition
- Has no persistent state as a whole

While the use of an object oriented programming model is not required in order to implement a component, Department of Education will standardize on the use of an object oriented programming language to implement components. Where possible the standard object oriented programming language will be Java. However, it is acceptable to use C++ in cases where only C / C++ Application Programming Interfaces (API) are available.

3.5. Standard Component Architectural Patterns

Viable software systems are developed according to some overall structuring principle. These principles are described using an *architectural pattern*. An architectural pattern establishes a fundamental structural organization on which to base a software implementation. It provides a framework in which a particular architectural problem can be solved. Architectural patterns are templates for technical architectures. The selection of an architectural pattern determines the strategy for developing a software system. The following are the architectural patterns which are described in further detail in the Conceptual Architecture (Volume 1) of the ITA detailed design document.

- Distributed N-Tier Client Server Architecture
  - N-Tier Architecture
  - Client Server Architecture
- Distributed System
- 'Model View Controller' Architectural Pattern
- Fat Client vs. Thin Client

3.6. The ITA Conceptual Architecture

The ITA is based on three core architecture domains within the SFA technical environment, which are targeted at reducing stovepipe systems, islands of technology and the need for customized point-to-point system interfaces. The three new domains include the Internet Domain, the Enterprise Application Integration Domain, and the Enterprise Data Domain. These three domains combined with the current Legacy Domain, make up the SFA enterprise and the ITA. The business rules, integrity checks and sequence of steps associated with a business function are implemented in a logical black box referred to as a 'service.' Services provide a set of published interfaces that allow participating applications to extend their business processes.

Together the four domains of the ITA provide the necessary infrastructure to implement a service-oriented architecture. However, in order to provide this infrastructure each domain must provide support for seamless integration between domain touch-points. Each domain must provide a set of interfaces or connectors that allow integration with the services of intersecting domains. For example, the Internet Domain must provide interfaces to the Data Domain in order to provide persistence for stateful business objects. The Data Domain must provide adapters to disparate data sources in order to feed the extract / transform / load process. The EAI Domain must provide connectivity to different legacy architectures and data sources in order to provide integration with existing SFA systems.

Figure 3 shows how the four domains overlap to provide seamless integration between domain services. The EAI layer is the glue for providing the integration between the domains where interfaces do not exist. The EAI layer provides a set of application adapters and communication services that can span domains thereby providing additional integration points between domains.

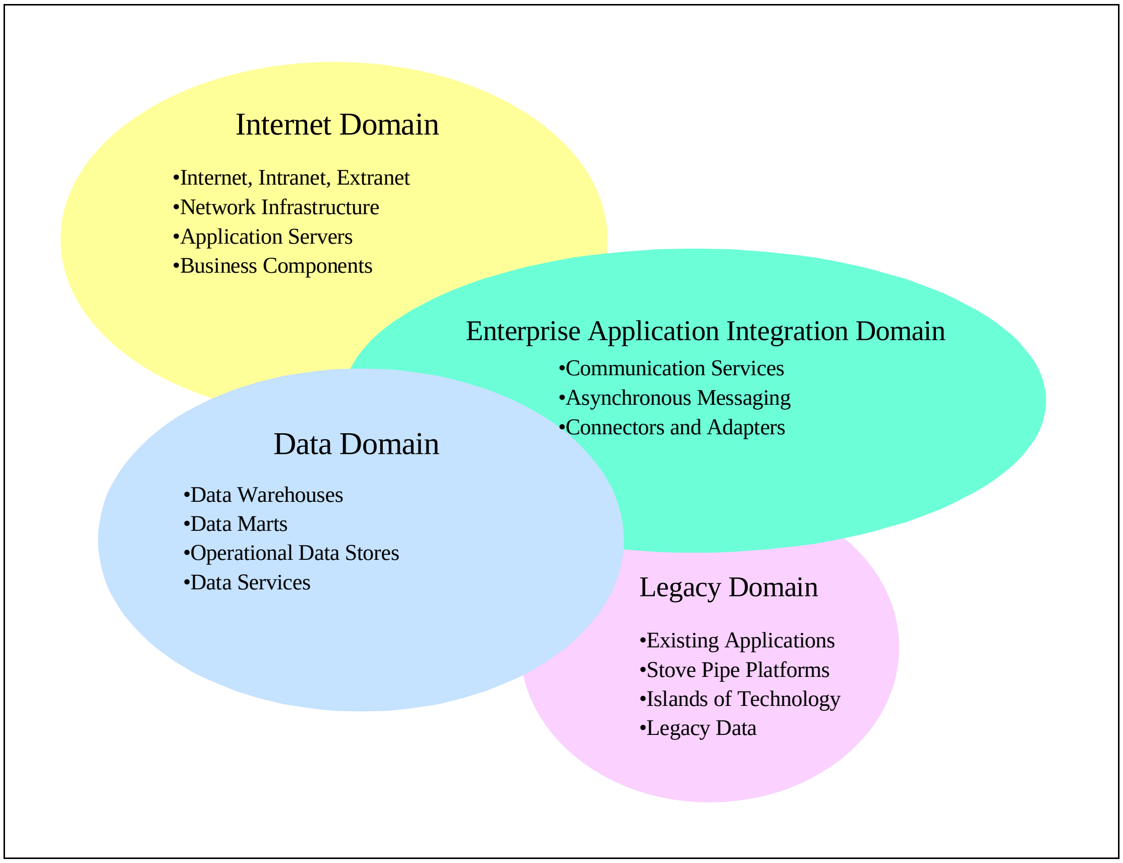


Figure 3 – The Four Domains of the SFA Architecture

The successful implementation of the ITA domains will provide the infrastructure for integrating Commercial-Off-the-Shelf (COTS) packages with other SFA systems. Using the integration interfaces supplied by the domains will make it possible to avoid the creation of system silos that are not part of the overall business solution. The ITA provides support for the integration of new COTS packages into the Internet, EAI or EDD. The ITA supports the integration of COTS packages that support the following domain interfaces or frameworks:

- Enterprise Java Beans (J2EE)
- CORBA Business Objects
- Work Flow Management
- Asynchronous Messaging (AMI, CMI, JMS, etc)
- Data Warehouse Analysis (ROLAP, MOPLAP, etc)

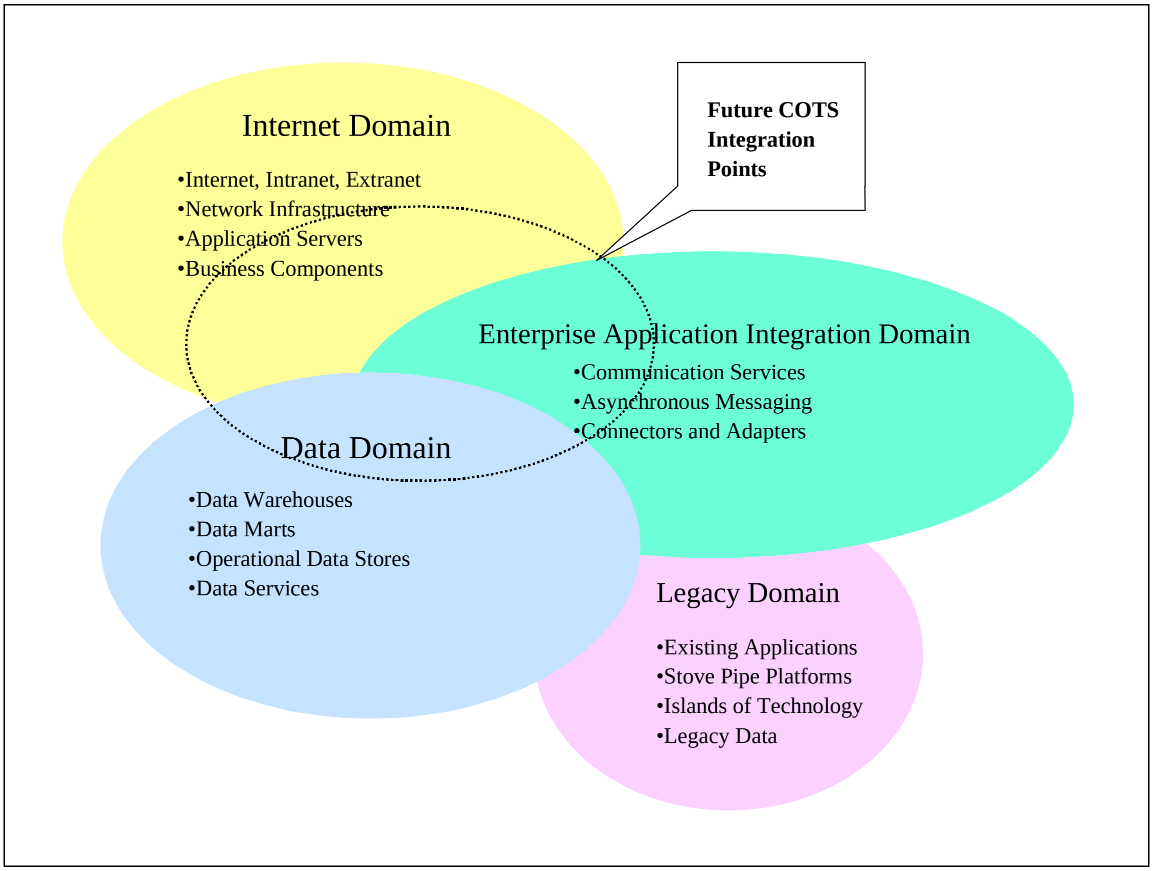


Figure 4 – Future SFA COTS Integration Points

3.6.1. Product to Domain Mapping

The Department of Education's SFA Modernization Task Order (TO) 4 defined the requirements for implementing a Service-Oriented Architecture. TO4 also selected the vendors and products that provide the functionality required for implementing the necessary functionality with each domain of the ITA. Table 2 provides a map for each selected product to the target domain according to function.

Table 1 – Integrated Technical Architecture Product to Domain Map

Product Type	Product Name	Function	Domain	Domain Interfaces
Web Server	IBM HTTP Server	- Thin Client Presentation - Render HTML	Internet	Enterprise Data
Web Application Server	IBM WebSphere Advanced Edition	- Servlets - JSPs - Web Security	Internet	Enterprise Data, EAI

Product Type	Product Name	Function	Domain	Domain Interfaces
Business Object Server	IBM WebSphere Enterprise Edition Component Broker	- EJBs - CORBA - Adapters	Internet / EAI	EAI, Legacy, Enterprise Data
Portal Server	Viador Portal Server	- Portal	Internet	
Search Engine	Autonomy Search Engine	- Web Spidering - Content Searching	Internet	Enterprise Data
Content Management	Interwoven TeamSite	- Content Management - Configuration Management	Internet	
Internet Load Balancing	IBM WebSphere Performance Pack	- HTTP Spraying - IP Redirection - Load Balancing - Caching	Internet	
Directory Server	Netscape LDAP Server	- Directory Services - Privilege Security	Internet / EAI	EAI
Message Oriented Middleware	MQSeries, MQSeries WorkFlow	- Asynchronous assured message delivery - Application Adapters - Business Process Management	EAI	Internet, Enterprise Data, Legacy
Message Broker	MQSeries Integrator	- Message Transformation and Routing	EAI	Enterprise Data, Legacy
Data Warehouse Management	Informatica	- Data Extraction - Data Transformation - Data Loading	Enterprise Data	Legacy
Data Warehouse Analysis	Microstrategy	- OLAP - ROLAP - MOLAP	Enterprise Data	Internet

4 Internet Architecture

The Internet Architecture (IA) supports the development, execution, and operation of web browser-based applications. It is composed of several independent components that integrate together to provide an overall net-centric capability. The integrated solution provides a maintainable, extendible, manageable solution that encourages and facilitates reuse and reduces development time by leveraging the architecture. It provides value by adapting existing legacy applications and standardizes application development, allowing for significant flexibility and cost savings.

The IA is the part of the Execution Architecture that provides the user dialog through a Web interface. Users of the ITA access the SFA systems through personalized information portals. These portals provide context sensitive access to legacy applications and enable users to search content by relevant subject. The IA bridges both the Internet and Intranet user community to the legacy SFA environment using the EAI Architecture and Data Warehouse Architecture. The Security Architecture insures that the Internet Architecture restricts and permits access appropriately for the user’s permission.

4.1. Internet Architecture Component Overview

The IA is composed of equipment and COTS applications. These components are specified and configured to provide superior performance and high availability.

The IA equipment consists of enterprise class servers. The servers utilized are Sun 3500 Solaris systems and other new technology (NT) based systems. The Sun 3500 Solaris systems are used for computationally intensive applications and applications requiring high availability. The NT based systems are used for applications that specifically require NT.

The IA applications consist of already existing Internet based applications augmented with other applications being deployed which interface with legacy enterprise SFA applications. The IA provides a way to integrate these applications through an information portal. Access to the already existing Internet based applications are through hypertext markup language (HTML) uniform resource locator (URL) links to the present Website. Other applications being deployed utilize the services of the IA as a function platform.

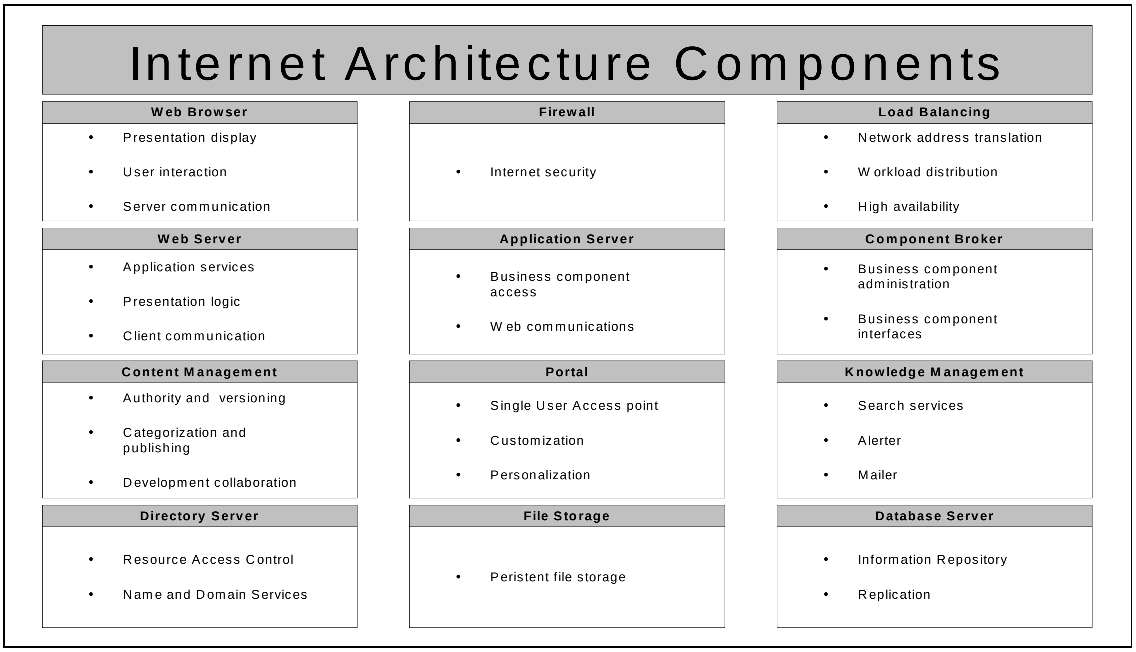
The IA COTS products consist of best-of-breed commercial applications that are integrated through standards-based API. Integrating these products through standards-based APIs enables other COTS products to be utilized when appropriate.

The IA framework is comprised of a set of components that provide the required services for a robust and secure Internet and Intranet environments. The principal functional components of the IA are listed in the following table.

Table 2 – Internet Architecture Components

Component	Description
Web Browser	Allow users to view and interact with applications and documents made up of varying data types, such as text, graphics, and audio. Provides support for navigation within and across documents no matter where they are located, through the use of links embedded into the document content.
Firewall	Protects sensitive resources or information attached to a network from unauthorized access. A variety of firewall implementations may be required at various levels within the SFA network model.
Load Balancing	Distributes IP traffic across a set of SFA application servers to achieve high availability and predicable performance.
Web Server	Enable SFA to manage and publish information and deploy network-centric applications over the Internet (public) and Intranet (private) environments
Application Server	Extend SFA capability by supporting net-centric applications as well as providing an application architecture for enabling the development and execution of common services across different business capabilities.
Component Broker	An Object-Oriented enterprise solution for distributed computing, providing a scalable, manageable environment for developing and deploying component based solutions.
Content Management	Manages the Website content set in a process-oriented fashion using configuration control methods. It provides content versioning and supports both structured and unstructured content formats.
Portal	Provides a customizable and personalized view as a single access point to a wide variety of heterogeneous data sources.
Knowledge Management	Provides an informative searching and retrieval capability for both structured and unstructured content. Information that can be searched includes but is not limited to documents, spreadsheets, HTML-based files, e-mail messages and electronic news feeds.
Directory Server	Act as a central data repository that simplifies communication and sharing of resources. It allows diverse applications, machines, and users (both inside and outside the enterprise) to access consistent information and services. This simplifies such tasks as electronic-mail addressing, maintenance of computing environments, and user authentication and authorization.
File Storage	Implements a secure and persistent network file system. Caching is utilized to ensure efficient resource utilization. High availability is achieved through replication of file systems and encapsulation of Storage Area Network (SAN) resources.
Database Server	Responsible for providing access to the operational data store (ODS). Maintains integrity of the data within the database and supports the ability to store data on either a single physical platform, or across multiple platforms.

The following diagram illustrates the principal components of the IA and lists the principal services they provide.



Internet Architecture Components

Web Browser
- Presentation display
- User interaction
- Server communication

Firewall
- Internet security

Load Balancing
- Network address translation
- W orkload distribution
- High availability

Web Server
- Application services
- Presentation logic
- Client communication

Application Server
- Business component access
- W eb communications

Component Broker
- Business component administration
- Business component interfaces

Content Management
- Authority and versioning
- Categorization and publishing
- Development collaboration

Portal
- Single User Access point
- Customization
- Personalization

Knowledge Management
- Search services
- Alerter
- Mailer

Directory Server
- Resource Access Control
- Name and Domain Services

File Storage
- Peristent file storage

Database Server
- Information Repository
- Replication

Figure 5 –Internet Architecture Components

4.2. Internet Architecture Technical Overview

The design of the IA is based on a critical set of technical requirements and assumptions. These requirements and assumptions determine the basic relationship of the individual IA components and the entire solution. The following table lists the principal requirements and assumptions, and presents their rational.

Table 3 – IA Requirements and Assumptions

Technical Requirements and Assumptions	Rationale
Scalability	The SFA solution supports both horizontal and vertical scaling. Horizontal scalability is achieved through application replication and load balancing. Vertical scalability is achieved through server chassis selection that allows both processing and data capacity expansion.
Reliability	The SFA solution achieves high availability through application and asset replication. Clustering techniques are not required for the SFA solution, although specific opportunities may require the use of clustering or clustering-like solutions.
Performance	The SFA equipment , COTS product and application configuration is based on engineering estimates derived from observation of existing systems and commercial best-practice implementation patterns typical of systems with equivalent user communities.

Technical Requirements and Assumptions	Rationale
Security	The initial release of the SFA solution implements authentication via Viador and application specific means. Other applications and COTS products may utilize a Viador API to determine the authentication parameters. Subsequent releases of the SFA solution would implement a standards-based security architecture. Established commercial best-practice implements separate Internet and Intranet content via separate Websites. Although the information portal COTS product provides access control to content based on user assigned group, content isolation using separate assets and content partitioning is a superior defense against intentional and unintentional security penetration. Department of Education security policy prohibits any web-browser active (dynamic) components

4.3. Solution

The IA is implemented as a set of servers that are interconnected via the Virtual Data Center (VDC) local area network (LAN). The VDC environment supports the servers and provides for system management. Other VDC assets provide essential services such as storage and archive of storage.

The IA servers were selected and configured with redundant components that are essential for sustaining operation. These essential components are power supply, processor, storage, and input/output.

Reliable power is achieved by using redundant power supplies and supplying separate power to each power supply. The VDC is responsible for providing and maintaining separate sources of conditioned power to insure that a power interruption event is isolated to only half the power source and corresponding power supplies.

The use of symmetric processing insures that processor failure only degrades performance and does not disable the entire server. The selected server family supports modular processor components. The use of modular processor components within the same server family allows spare processors to be acquired and managed efficiently. The spare processors would be available to support recovery of any server. Recovery from a processor failure may require cycling the server but once cycled the server would be available.

System storage is implemented with storage that is installed within the system and with storage that is available via network. The system installed storage units provide for the operating system and the storage required for operation and management of the system. The system storage units are redundant and the stored information is replicated in order to survive storage errors and general storage failure. Network storage is utilized for bulk storage. Network storage provides reliable information retention and supports sharing of information across servers.

Survivable input/output is achieved by using redundant adapters and configuring access to redundant networks. Using redundant adapters and having access to redundant networks also enables input/output balancing. Failure of an adapter or a network may impact performance but the application should persist. Recovery from an adapter failure is possible by switching network connections to an available adapter. Recovery from a network failure should be transparent with automatic recovery.

Internet and Intranet connectivity is provided by the VDC. The VDC maintains redundant Internet connectivity through separate service providers and arranges for adequate capacity.

The server equipment utilizes the VDC LAN for connectivity between the servers and other VDC equipment such as network storage and legacy systems. Connectivity between the servers for application specific traffic may require the deployment and configuration of dedicated sub-networks. The IA implementation is phased to support an initial application deployment with limited user community and a subsequent deployment to address other applications and users.

The following diagram illustrates the initial deployment of the IA equipment and the allocation of COTS products to the equipment.

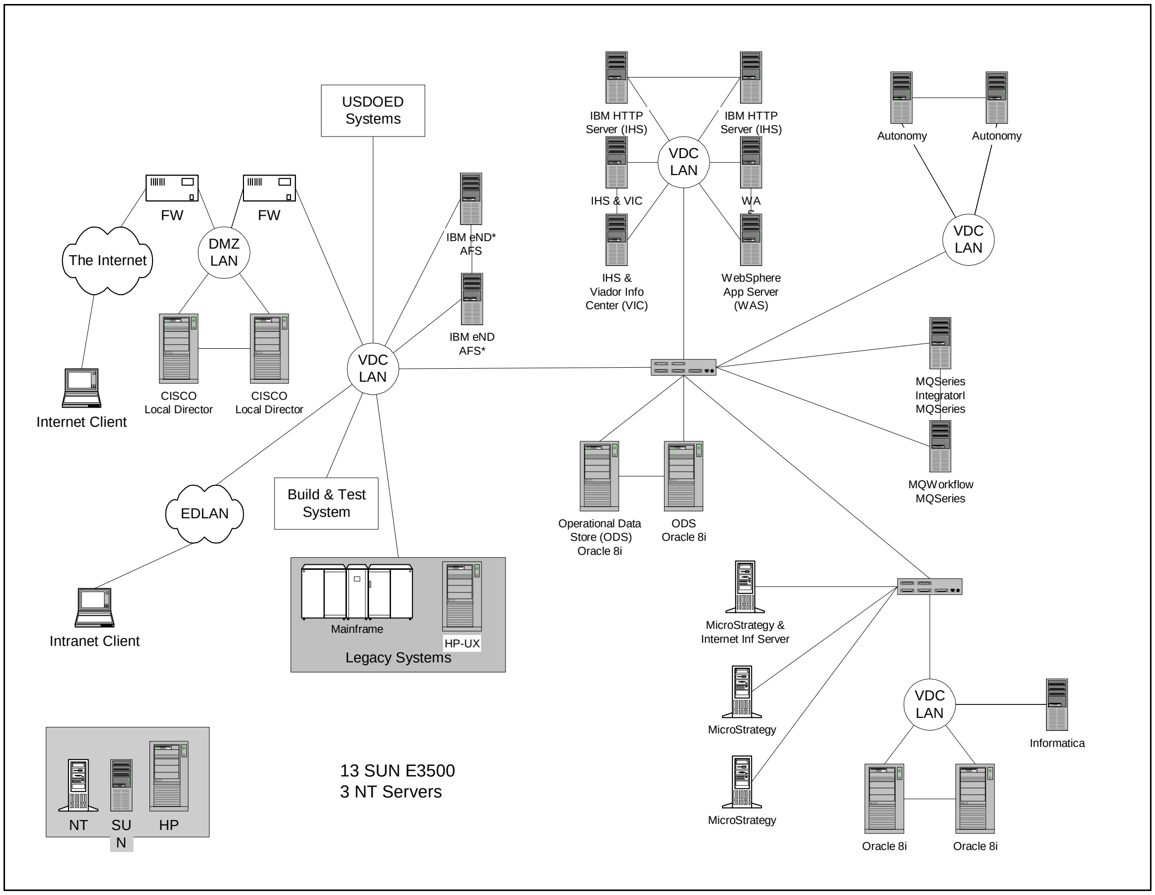


Figure 6 – IA Implementation

The IA consists of 12 architectural components. The components partition the functional capabilities of the IA and provide a method to identify suitable COTS products to implement the IA. Some of the components are supplied by the DOE or are native capabilities of the

VDC. The following table lists each of the 12 architectural components of the IA, the COTS product or products selected to implement the component, and the respective source (IA or VDC) of the component. The specific version of each COTS product is specified in the table that follows.

Table 4 – IA Component COTS Products

Component		Product	Source
1	Web Browser	Microsoft Internet Explorer, Netscape Navigator, or Lynx	N / A
2	Firewall	Check Point FireWall-1	VDC
3	Load Balancing	IBM eNetwork Dispatcher	IA
4	Web Server	IBM HTTP Server	IA
5	Application Server	IBM WebSphere Application Server Enterprise Edition	IA
6	Component Broker	IBM Component Broker	IA
7	Content Management	Interwoven TeamSite	IA
8	Portal	Viador Portal Suite	IA
9	Knowledge Management	Autonomy Knowledge Suite	IA
10	Directory Server	Netscape Directory Server	VDC
11	File Storage	IBM AFS	IA
12	Database Server	Oracle 8i	VDC

4.4. Internet Architecture Technical Services

The IA components establish and provide functional services that enable the SFA applications. These services form a rich and efficient service-oriented layer and are the preferred methods for applications to invoke and utilize the IA. The following subsections, organized by architectural component, identify and functionally describe the services.

4.4.1. Web Browser

The Web Browser component is the user access mechanism to the IA. The component is a standard COTS product that implements the hypertext transfer protocol (HTTP) protocol, and renders HTML. The Web Browser services provide retention of the link connection, i.e., document physical location, and mask the complexities of that connection from the user. The following table identifies and describes the services provided by this component.

Table 5 – Services – Web Browser

Service	Functional Description
Server Communication	Utilizes standard protocols to establish a data connection to the server, transfer data, and terminate the connection.

Service	Functional Description
Communication Security	Interoperable methods of securing the communications channel between the client and the server using secure protocols, e.g. SSL.
Presentation Services	Renders text, graphics, and other user interface components. Utilizes HTML as the content description language.

4.4.2. Firewall

The Firewall component protects SFA resources against direct and indirect intrusion. Access is managed using policy-driven restrictions on network connections, protocols, and data formats, and application-driven restrictions on data exchanges by applications and individuals.

Table 6 – Services – Firewall

Service	Functional Description
Packet Filtering	Protocol-based services check the address portion of data packets to determine the desired destination and intent. Administrators can block certain combinations that are categorized as unauthorized.
Proxy Services	Establishes a shielding or screening of the server address which is typically placed between the Internet router and the private network assets to be protected. Proxy servers shield users from knowing the specific addresses of servers within the private network.

4.4.3. Load Balancing

The Load Balancing component distributes workload across a set of applications and associated servers. The following table identifies and describes the services provided by this component.

Table 7 – Services – Load Balancing

Service	Functional Description
Network Address Translation	Distributes URL and other client-server workload across a set of servers using network address translation. Load balancing is achieved using various algorithms that account for resource availability in order to insure reasonable response time. Virtual address support allows the client request to be distributed across a set of servers and each client is only aware of the virtual address.

4.4.4. Web Server

The Web Server component manages document requests in formats such as HTML, Portable Document Format (PDF), etc.

The following table identifies and describes the services provided by this component.

Table 8 – Services – Web Server

Service	Functional Description
Client Communication	Uses HTTP to establish a data connection to the server, transfer data, and tear-down the connection.
Communication Security	Interoperable method of securing the communications channel between the client and the server using SSL.
Dynamic Page Services	Utilizes JSP to execute commands that are embedded in the presentation data to allow for run-time binding of presentation and data.
Application Services	A servlet based method to execute commands that interact with external systems and components, returning data to the requesting client.

4.4.5. Application Server

The Application Server component provides access to legacy systems, databases, and other application servers through a reusable and consistent application architecture. Applications are supported within a standards-based open development environment that provides the ability to use object-oriented technologies. The following table identifies and describes the services provided by this component.

Table 9 – Services – Application Server

Service	Functional Description
Run-Time Services	A Java / CORBA compliant application environment. Java Virtual Machine (JVM) compliant with Java 1.2 or greater.
Application Services	Programmatic specification of business logic in a reusable, component manner using an Enterprise JavaBeans implementation and behavior model.
Database Services	A JDBC API to database systems, regardless of database vendor.
Transport Services	Guaranteed message delivery service between components. A JMS / Javamail unified programming interface to external email and messaging servers, regardless of server vendor.
Directory Services	A JNDI facility used to discover remote network resources.
Transaction Management	A JTA based service allows multi-step processes to succeed or fail as an atomic unit.
Object Communications	Standard approach for objects to call the methods of other objects and RMI / IIOP ability to communicate across the network to CORBA objects.
Data Typing and Encryption	A JAF facility that helps components determine the data type of an arbitrary data stream, then encapsulate that stream into a known object format.

4.4.6. Component Broker

The Component Broker (CB) component provides implementation services to business objects or enterprise beans. Some of these object services are administrative in nature and their behavior is controlled by qualities-of-service configured through the management tools. Other services are presented to business object implementers as interfaces, and others are built into the infrastructure and work on behalf of the business logic. The following table identifies and describes the services provided by this component.

Table 10 – Services – Component Broker Component

Service	Functional Description
Concurrency Control Service	The Concurrency Control Service consists of a set of interfaces that allow an application to coordinate access by multiple transactions or threads to a shared resource. When multiple transactions or threads try to access a single resource at the same time, any conflicting actions are reconciled so that the resource remains in a consistent state.
Event Service	The Event Service defines a channel between multiple objects which defines their roles and allows them to communicate asynchronously. There are two defined roles: supplier objects and consumer objects. Suppliers produce events, while consumers process events.
Notification Service	CB's Notification Service contains event channels that act as supplier and consumer objects. These event channels allow multiple suppliers to communicate with multiple consumers asynchronously and without confusing the many low-level details within the objects.
Externalization Service	The Externalization Service provides a mechanism by which objects are able to save and restore their state in a non-object form. This allows the object's state to exist independently of the object itself.
Identity Service	CB derives an object identity from relative information that positions the object within its container, server, host, and domain. This information can be used within the CB Managed Object Framework to uniquely identify each object from any other object in the distributed system.
Life Cycle Service	A Life Cycle Service provides operations for creating, copying, moving, and deleting objects in a distributed environment. The Life Cycle Service in CB provides a level of abstraction between the client program creating an object and the determination of the location where that new object will exist.
Naming Service	The CB Naming Service allows you to create naming hierarchies so you can easily locate objects. In conjunction with other services, clients can navigate through different naming context trees to locate specific objects. CB Naming Service handles both absolute and relative paths.
Security Service	The Security Service is used primarily to prevent end users from accessing information and resources that they are not authorized to use. This predominantly covers distributed business objects, but by extension includes any of the information and resources from other non-object-oriented or non-distributed sources used by those business objects.
Transaction Service	The Transaction Service enables programmers to implement transactions by using standard object-oriented interfaces in a distributed environment. CB uses the Transaction Service to ensure that each application has correctly grouped the updates in the transaction so that the data is always updated consistently.
Session Service	The Session Service provides detailed information for applications in a distributed object environment to control the extent of a session and the application profile and arbitrary session properties that are relevant within the scope of that session. The scope of the session is defined to exist between the point when the session is started and the point when the session is ended.

Service	Functional Description
Query Service	The Query Service enables you to find objects in a CB collection based on a set of conditions described with an object-oriented structure query language (OOSQL). The OOSQL enables you to describe complex search criteria. It is a extension of SQL with features for handling object collections, object attributes, and methods in query statements.
Cache Service	The Cache Service enhances concurrency and performance by supporting optimistic and pessimistic caching of data. In optimistic caching, frequently used data is cached in the memory of the CB server and not reread from the database on each transaction. Cached data is invalidated based on a time-out value.
Workload Management	The Workload Management capability allows the CB run time to dynamically allocate an application server to process a request. As more clients use an application, the amount of work increases and the load on the servers increases. The key to workload distribution in CB is the use of a server group to define multiple application servers with a common configuration.

4.4.7. Content Management

The Content Management component manages Website content delivery from the development environment to the production environment. The following table identifies and describes the services provided by this component.

Table 11 – Services – Content Management

Service	Functional Description
Authoring	Allows users to associate and launch development applications against the content managed by the component.
Versioning	Maintains versions of each individual Website content artifact. The individual content versions are associated with Website configurations or releases.
Categorization & Publishing	Manages groups of content artifacts according to user defined criteria and supports publishing of these content artifacts.
Development Collaboration & Workflow	Provides process control and related methods that support collaboration between personnel in the development community and production community. The collaboration and workflow utilities provide a methodical way to insure that content change is appropriately authorized.
Integrates Multiple File Types	Any file type is supported. The Interwoven product is not aware of or dependent on the file type.
Summarization	Produces a summary report of a configuration or release and the Website content artifacts that were delivered from the development environment to the production environment.

4.4.8. Portal

The Portal component provides a customizable and personalized interface as a single access point to a wide variety of heterogeneous data sources including Website content, documents, and existing applications. The following table identifies and describes the services provided by this component.

Table 12 – Services – Portal

Service	Functional Description
Single Point Of Access	The portal provides a single interface in which to access a wide variety of heterogeneous data sources within SFA. These heterogeneous data sources can consist of the following: structured data such as content documentation, unstructured data such as intranet and Internet Web content and existing enterprise applications.
Customization	The portal can be customized to provide access to a range of data sources and applications based on the users and their roles. These roles can be defined by the SFA system administrator and then applied to the categories of SFA users such as students, schools, etc.
Personalization	The portal can be personalized by selecting from the catalog of data sources and applications accessible according to the assigned roles of the SFA users. This allows the SFA users the ability to quickly locate required information and filter extraneous information.
Authorization	The portal provides usage authorization to control the level of granular access an SFA user has to the portal itself. This is organized by individuals, roles, or groups as defined by the SFA system administrator.
Authentication	The portal provides authentication as the process of uniquely identifying a specific SFA user and maintaining the accessibility of information as defined in the customization performed by the SFA system administrator.

4.4.9. Knowledge Management

The Knowledge Management component provides the information search and retrieval capability. This component offers various search types on different data groups as unstructured digital information, structured data, word processing documents, HTML-based files, e-mail messages and electronic news feeds. This product is able to support a thesaurus query if a thesaurus is loaded into the environment. The following table identifies and describes the services provided by this component.

Table 13 – Services – Knowledge Management

Service	Functional Description
National Language Search	A search capability specified as a subject / verb criteria.
Boolean Search	A structured search capability specified as a Boolean or logical expression criteria.
Proximity Search	A search capability based upon a text delta algorithm.
Proper Names Search	A search capability restricted to proper names.
Simple keyword	A search capability based on a key text string.
Query Search	A search capability based on queries.
Bracketed Boolean Search	A search capability based on a cobination of Boolean expression criteris.
Mailer	A scheduled process that queries the DRE and sends the results to the users either as a list of links to content in a single message, or the content itself in a separate messages.

Service	Functional Description
Alerter	A feature that will direct a large flow if information to those individuals who have an interest in it.

4.4.10. Directory Server

The Directory Server component manages information common to applications, individuals, and groups of individuals. The following table identifies and describes the services provided by this component.

Table 14 – Services – Directory Server

Service	Functional Description
Name Services	A logical component of directory services provided to create a logical "pronounceable" name in place of a binary machine number. These services are used by other communication services such as file transfer, message services, and terminal services.
Domain Services	Provide a mechanism by which various nodes are recognized. These services use the domain portion of an address to transport the data to the corresponding node. Therefore, Domain Services are functions that track and recognize different logical organizations and then map them to physical resources as tracked by the Naming Services.
Single Sign-on Services	Supports a single sign-on capability by providing a common user authentication repository and a standards-based access method. Single sign-on is actually implemented in association with other products, but the essential framework is LDAP.
Personalization Preferences	Individual and groups of individuals may be associated with preferences. This capability manages these preferences on both a global basis and per application
Authentication	User sign-on is verified using a password. User passwords are administered with security controls.
Access Control	User access to applications and specific files may be controlled. Access controls may be allocated and enforced for individuals or for a group of individuals.

4.4.11. File Storage

The File Storage component provides an IA file system hierarchy that utilizes the VDC Storage Area Network (SAN). The Andrew File System (AFS) is used to achieve performance and high availability. The following table identifies and describes the services provided by this component.

Table 15 – Services – File Storage

Service	Functional Description
Fileserver	Handle requests at the file and directory level.
Volume Management	Handles operations at the volume level.
Consistency Checking	Checks the system for internal consistency and repairs errors it finds.

Service	Functional Description
Authentication	Responsible for maintaining the Authentication Database.
Access Control	Responsible for maintaining the Protection Database.
Cache Management	Responsible for maintaining the database and for providing the Cache Manager with information about volumes and volume location.
File System Backup	Responsible for maintaining the Backup Database and for providing an interface to the AFS Backup System.
File System Synchronization	Responsible for transferring information from System Control Machines (SCM) and Binary Distribution Machines (BDM) to other AFS servers.
Directory Distribution	Responsible for distributing the contents of a specified directory.
Time Synchronization	Synchronizes an AFS server's clock with the clock on another machine.

4.4.12. Database Server

The Database Server component provides a consistent relational interface to information contained in a database. The component supports high performance storage and retrieval of structured data. The following table identifies and describes the services provided by this component.

Table 16 – Services – Database Server

Service	Functional Description
Storage Services	Manage data physical storage. These services provide a mechanism for saving information so that data will live beyond program execution. Data is often stored in relational format (an RDBMS) but may also be stored in an object-oriented format (OODBMS) or other formats such as IMS, VSAM, etc.
Indexing Services	Provide a mechanism for speeding up data retrieval. In relational databases one or more fields can be used to construct the index. So when a user searches for a specific record, rather than scanning the whole table sequentially the index is used to find the location of that record faster.
Security Services	Enforce control regarding which records authorized users can view and edit, and which functions they can execute. Most database management systems provide data access control at the database, table and row levels, and execution control for stored procedures, database functions, etc. to specific users and groups.
Access Services	Enable an application to retrieve data from a database as well as manipulate (insert, update, delete) data in a database. SQL is the primary approach for accessing records in today's database management systems.
Replication Services	Support an environment in which multiple copies of databases must be maintained.

5 EAI Architecture

The EAI system is part of the Executive Architecture for the Department of Education SFA system as part of the Modernization Blueprint. EAI is a set of technology services that enables the sharing of processes and data of disparate systems to support end-to-end business processes. The EAI Architecture enables the many "stovepipe" applications to exchange information via common, reusable methods and infrastructure.

EAI will allow the SFA program for the Department of Education to integrate new web-based applications with existing back-end systems, while at the same time, providing a means to migrate away from reliance upon existing legacy systems.

EAI provides capabilities that will allow for the integration of web-based applications, the Data Warehouse environment, COTS packages, and existing legacy systems within the SFA technical environment.

The EAI architecture provides the following technical services:

- Communications Middleware
- Transformation and Formatting
- Application Connectivity
- Business Process Management

5.1. EAI Architecture Solution

The EAI Architecture solution for the ITA is based on the International Business Machines (IBM) MQSeries product family. A specific MQSeries application or component will provide each of the high-level EAI technical services.

Communications Middleware → MQSeries Messaging

Transformation and Formatting → MQSeries Integrator.

Application Connectivity → Adapters and MQSeries Bridges

Business Process Management → MQSeries Workflow

The diagram below depicts the EAI Architecture and the relationship between the MQSeries products. The MQSeries Adapters and Bridges are used as interfaces with external systems and are not shown as part of this diagram.

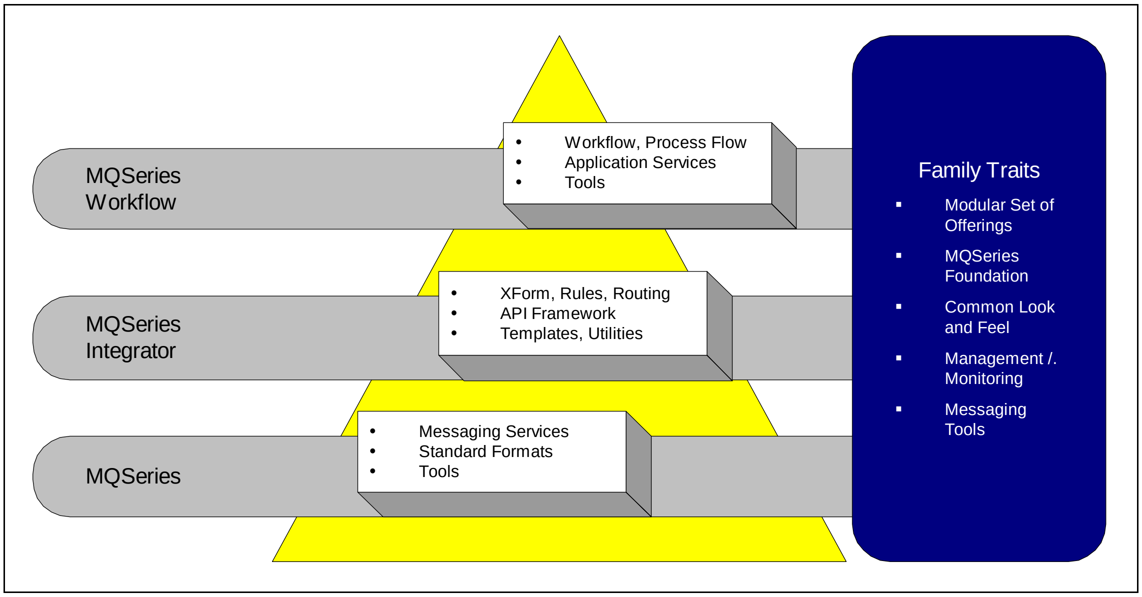


Figure 7 – EAI MQSeries Product

5.2. Communication Middleware

The Communications Middleware component provides the architecture that implement various messaging models and route messages according to message content and context. These services provide the connection among disparate resources, as well as security, queuing, and the functionality to reconcile network protocol differences.

Communications middleware:

- Directs the flow of messages among applications
- Supports both synchronous and asynchronous communications
- Routes messages to applications based on message subject and / or content
- Provides services via message brokers, Object Request Brokers (ORB), or message queues

5.3. Transformation and Formatting

The Transformation and Formatting layer is responsible for the conversion of data and message content and syntax to reconcile the differences between data from multiple heterogeneous systems and data sources. This layer is responsible for maintaining the information structure of the messages passed between systems and their meaning in a format that can be comprehended by another application.

The transformation and formatting layer supports:

- Message protocol and format transformation

- Syntactic translation of one data set into another. (Example: translation of date formats, 01 Aug 1999 -> 19990801)
- Semantic translation of data based on underlying data definitions or meaning. (Example: conversion from the English system to the metric system)

5.4. Application Connectivity

The Application Connectivity layer provides reusable, non-invasive connectivity with legacy systems and external databases.

The Application Connectivity layer provides:

- Pre-built application adapters to access legacy systems and databases
- Connection managed to and from source application
- Connectors to common technologies such as ORBs, support for CORBA, EJB, etc.

5.5. Business Process Management

The Business Process Management layer is responsible for the definition and management of cross-application business processes across the enterprise and between enterprises. These services enable the communication not just of data, but also of the business process context of the data being sent to another application.

Business Process Management provides:

- Centralized visibility and control of multi-step business processes traversing multiple applications
- Real-time analysis capabilities
- Workflow-like coordination of multi-step processes
- Transactional control
- Process state information maintained to support rollback processes
- Graphical tools and metadata to define processes and rules

6 Data Warehouse Architecture

The Data Warehouse volume provides the detailed design specifications necessary to build and maintain the Data Warehouse Architecture (DWA) for the SFA.

Building a data warehouse is an iterative process ultimately driven by project requirements. As such, the initial release of the Data Warehouse Architecture provides the design details needed to implement the DW within the context of the Chief Financial Office (CFO) Datamart and Central Data System (CDS) Retirement projects. These are currently the only two SFA data warehouse projects scheduled for Release 1 of the ITA. However, the designed infrastructure will not only support these current business requirements, but also will be fully scalable as SFA data warehousing needs expand and grow over time.

6.1. Data Warehouse Architecture Domain

The data warehouse is defined as a subject oriented, integrated, time-variant, non-volatile collection of data used to support the decision making process of the SFA. Data in the data warehouse is generally a copy of transaction data (though non-transaction data may also be included) and is specifically structured for querying and reporting. The data warehouse will reside on disks and servers that are separate from the SFA's transaction processing systems and be configured specifically to facilitate speedy and accurate querying and reporting.

6.2. Data Warehouse Process Flow

The data warehouse process flow defines the flow between the various applications or "tools" that comprise the Data Warehouse architecture. Figure 8 provides a high-level illustration of this process flow.

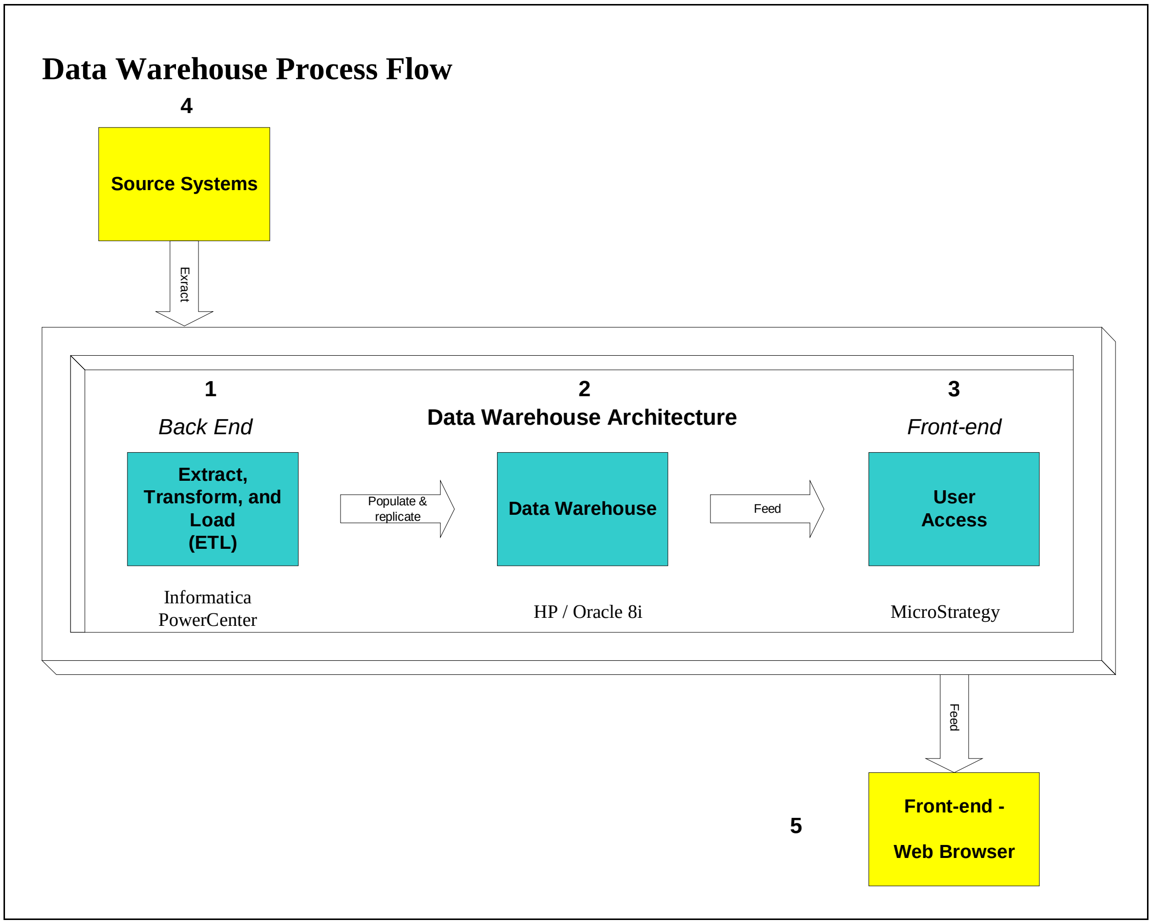


Figure 8 – Data Warehouse Process Flow

The DWA (Figure 8) is comprised of the following three components:

1. *The Extraction, Transformation, and Loading Process (ETL)*, also known as the “back-end.” This is provided by Informatica’s PowerCenter 1.7.

2. The *Data Warehouse*, which resides on an Oracle 8i Relational Database Management Systems (RDBMS).

3. *User Access*, also known as the “front-end” is provided by MicroStrategy 7.0 Platform and which enables the user interactive reporting capabilities such as drilling, pivoting, and report creation.

The following components reside outside the DWA:

1. *Source systems*, which are the operational system of record whose function is to capture the transactions. In the SFA environment these will primarily consist of legacy systems including DB2 mainframe data, Informix databases, and flat files.

2. The *Web Browser* is provided by MicroStrategy Web and enables the user the same interactive reporting capabilities through a web browser rather than the client application.

6.3. Data Warehouse Process Flow Details

Figure 9 “drills down” the DWA into a detail to provide greater detail.

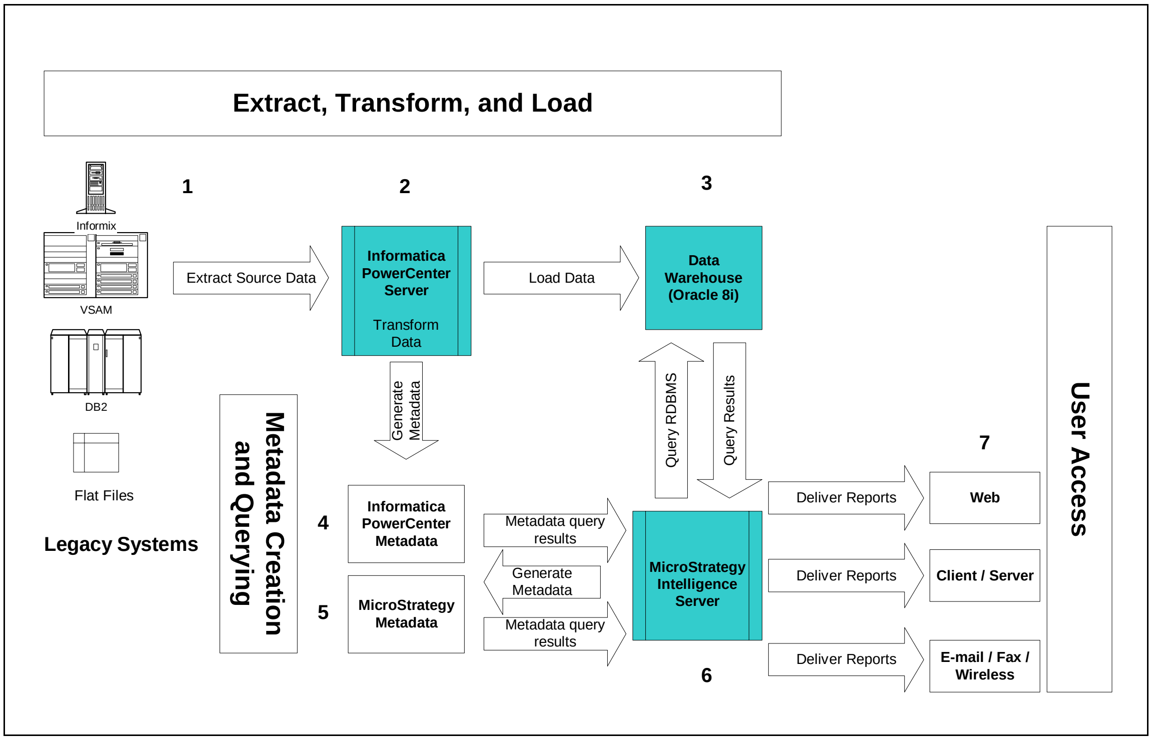


Figure 9 – Data Warehouse Process Flow Details

Extract, Transform, and Load - The ETL process consists of several steps. First is the extraction or reading the source data. Once data is extracted, numerous transformation steps may be undertaken, including cleansing erroneous data, purging unnecessary fields, combining sources, and building aggregates. Loading, the final step in the ETL process, is the activity of replicating dimension and fact tables and presenting them to the data warehouse via bulk loading facilities.

User Access - Once data is loaded into the Data Warehouse, User Access can occur. In the User Access module, the MicroStrategy server processes query requests initiated by the front-end web browser, translates these requests into Structured Query Language (SQL) statements and returns results by delivering reports to the user via the web-browser.

Metadata Creation and Querying – Metadata is created during both the ETL and User Access process. Both the Informatica and MicroStrategy applications create metadata that is stored in their respective repositories.

1) **Legacy Systems** – These will provide the primary source data and will include VSAM files, DB2 Mainframe data, and flat files.

2) **Informatica PowerCenter Server** – is the actual engine that provides the data extraction, transformation, population services as well as creation and management of Informatica Metadata.

3) **Data Warehouse** – the Oracle 8i RDBMS where the data physically resides.

4) **Informatica PowerCenter Metadata** – will provide the technical metadata regarding data sources, field names, target tables, etc. primarily aimed at developers and Database Administrator (DBA) resources.

5) **MicroStrategy Metadata** – Provides the business metadata, generally aimed at the business users seeking file structure definitions, business rules.

6) **MicroStrategy Intelligence Server -** is the middle-tier between the user applications and the data warehouse providing report cache management, analytical functions, job prioritization, and thread management.

7) **Web / Client Server / E-mail / Fax / Wireless** – the various outputs available for reports created by the MicroStrategy 7 Platform.

7 Security Architecture

The Security Architecture (SA) volume within the ITA Detailed Design Document provides an overview of the security requirements, gaps between the current environment and the requirements, and architectural recommendations for the Release 1 of the ITA. Applications that will leverage this release of the ITA include: Schools Portal, Information for Financial Aid Professionals (IFAP), and Intranet Release 2.0.

The Security Architecture volume includes:

- Security Framework Overview
- Analysis of the Current Environment: An assessment of SFA's current security framework for the Release 1 applications (IFAP, School Portals, and Intranet Release 2.0) as well as future releases of SFA applications.
- Proposed Architectural Designs: A proposed security framework based upon Common Operating Environment (COE) and Internet Security Standards Task Order 4 Deliverable 4.1.3.
- Gap Analysis of Current Infrastructure: An analysis of the security needs required to fulfill the proposed architectural design both short and long term.
- Architectural tool recommendations, options, and implementation examples.

Systems that were evaluated for Release 1 of the ITA include:

- IFAP
- Schools Portal
- Intranet Release 2.0

Systems that were evaluated for the post -Release 1 included:

- Campus Based Systems (CBS)
- Central Processing System (CPS)
- Direct Loan Consolidation System (DLCS)
- Direct Loan Origination System (DLOS)
- Direct Loan Servicing System (DLSS)
- Federal Family Education Loan (FFEL)

- Multiple Data Entry (MDE)
- National Student Loan Data System (NSLDS)
- Post-secondary Education Participants System (PEPS)
- Recipient and Financial Management System (RFMS)
- Title IV Wide Area Network (TIVWAN)

After reviewing the pertinent documents, meetings were held with members of the SFA Chief Information Office (CIO) executive team, channel owners, VDC support personnel, and various hardware / software vendors who provide services and support for SFA.

From the information gathered from these resources as well as internal knowledge capital, the SFA Security Architecture document was developed. The document describes the architectural requirements, gaps, and components of the SFA Security Framework, which addresses several security paradigms: fine-grained access control, authorization, authentication and single sign-on potential. The combination of these functions within a single entity, conceptually referred to as an enterprise security portal, provides SFA with a secure, reliable, and available framework to its varied applications, Web sites, and databases.

7.1. SFA Security Framework Recommendations Summary

Current Security Assessment: The current SFA network architecture provides a satisfactory level of perimeter-level protection for the ITA systems for Release 1.

Overall Recommendation for Release 1: The Release 1 systems should continue to be enhanced from a security perspective. There are certain security features, which should be implemented as these systems proceed and these features apply to the overall SFA architecture as well.

Implementation of the following is recommended:

- Intrusion detection systems are needed at both a network and host level. Regardless of the protection provided at a firewall level, systems should be monitored for intrusion. Firewalls present a first line of defense but no single element of security provides comprehensive protection. As is the case in physical security if someone attacks and penetrates a steel door with three deadbolt locks, an alarm should alert the proper group of the compromise. This scenario applies the same to the network perimeter and internal systems housing critical applications. If the firewall experiences a compromise, an automatic alarm is triggered and a log of all activity is made available by network intrusion detection systems. Specific Vendor Product Recommendations are provided in Appendix A (Rows 1-4) of Volume 5 – ITA Security Architecture Design Document.

- Policy compliance assessment tools are recommended. These automated tools provide security configuration compliance baselines and measure for deviations from that baseline at scheduled intervals. Reports provide system administrators with detailed recommendations needed to resolve problems found. Vendor Recommendation provided in Appendix A (Rows 5-6) of Volume 5 – ITA Security Architecture Design Document under the Information Security Core Technology Status section.

- It is highly suggested that all systems scheduled for Release 1 have a thorough vulnerability assessment conducted after code freeze, but prior to production rollout. Currently, it appears that the Modernization Partner and CSC are planning to conduct these tests. Tests should include analysis of application code as well as host systems and networks. Vendor recommendation provided in Appendix A (Row 3) of Volume 5 – ITA Security Architecture Design Document.

Overall Recommendation – Post- Release 1: SFA should continue to build active defenses from intrusion for all systems, starting with the systems being delivered for Release 1.

In addition to the recommendations for the Release 1 applications, the following tasks should be accomplished post-Release 1:

- A centralized monitoring, log-collection and reporting solution to support real-time intrusion detection and overall security management is needed for the SFA Security Infrastructure. Automated tools exist which can collate information from a variety of sources (firewalls, intrusion detection devices, policy compliance tools, etc), facilitate alert mechanisms, and provide summary reporting. This alleviates the extensive man-hours required to sift through log files, which ultimately delay detection of network penetration until well after the fact. Such a system supports near real-time reporting, versus post incident reporting. Vendor Recommendations provided in Appendix A (Rows 2, 3, 8-11) of Volume 5 – ITA Security Architecture Design Document.

- An Information Security standard of configuration is required for each major technology component (e.g., Sybase, Oracle, NT, Solaris, IIS, etc). These baselines provide product specific details to developers and architects to for secure configuration of SFA systems and allow establishment of technical security parameters on each system. The result is a more common system level build, testing structure, configuration management structure, and stronger overall data integrity. Vendor Recommendations provided in Appendix A (Row 10) of Volume 5 – ITA Security Architecture Design Document.

- An enabling Security Framework architecture will allow SFA to move away from the current "hairball" development methodologies. Such architecture will provide an infrastructure supporting Information Security initiatives during the application development cycle and consists of several minimum components to include:

  - Lightweight Directory Access Protocol (LDAP)

  - Privacy Management

  - Central Risk Management

  - UserID Management

  - API / Toolkits for SFA integration to MQ, Websphere, Java, and CORBA and also support a Portal based architecture. Vendor Recommendations provided in Appendix A (Row 15) of Volume 5 – ITA Security Architecture Design Document.

- It is recommended that SFA implement a formal Risk Assessment program. The purposes of this program is to:

  - Provide business managers with a process to integrate security risk management into the decision support process for business operations.

  - Implement a business-risk based approach to identifying and assessing information security risks in the terms of the impact on business operations.

  - Provide the business manager a basis for determining what controls are needed and what level of resources can be expended on controls.

7.2. Security Framework Overview

The SFA is moving towards allowing its customers and its partners high-speed, secure system access over the Internet. In order to make this happen, the architecture that supports this access must provide confidentiality, identification, authentication, authorization, data integrity, accountability, and non-repudiation for all transactions initiated.

The Security Framework is a usable and comprehensive security overview. This Security Framework should be thought of as a conceptual structure used to frame the security related work to be designed and implemented.

The Security Framework is used to help SFA understand what security components may be required and how the components fit together. Based on the inventory of components and the description of their relationships, the optimal solutions will be applied.

Multiple instances of security frameworks may be necessary to facilitate business needs. The number and location of these infrastructures will be driven by business and institutional needs enabled by security, performance, and quick reaction capability. If more than a single framework is required, the directory structures for each framework can be replicated across the infrastructures. In the case of an infrastructure failure, traffic can be routed to another framework providing redundancy of operations transparent to the end-user. Each framework can be configured for high availability sharing processing loads across infrastructures. Where multiple infrastructures are implemented to provide a common set of security services, the virtual aspect of framework design can be used to balance the load across diversely located SFA systems thereby achieving optimal utilization of SFA resources and reducing capital investment.

The components of an Enterprise Security Framework will consist of:

- Business Assets - represents what needs protection, and is the target of all information security efforts. The SFA Security Framework will contain all the necessary hardware and software to secure most SFA resources including legacy applications (client server and mainframe). The framework should furnish the necessary features that make the secure implementation of Business-to-Customer, Business-to-Business, and internal based systems more efficient and systematic.

- Risk Management - analyzes the value of business assets, the cost to protect the assets, the level of protection required, and discovers the threats and vulnerabilities that must be addressed through the security strategy. The Security Framework provides event monitoring, logging and detection of multiple types of activities. Implementation of the Security Framework allows detection when an event occurs that violates the system's security policy, generates alerts, and allows administrators to determine how to respond to the attempt.

- Security Strategy - defines the approach and direction SFA is taking to secure and enable the Business Assets in line with the Risk Management approach. Within industry and government most major systems development, communications, and financial transactions are moving to an Internet. No longer is it enough to provide basic security commodity services (firewalls, secure routers, virus protection, etc) which block and disable, but it is also crucial to be able to provide enabling services to allow all financial institutions, academic organizations, and individual users to securely access SFA resources over the Internet. The focus of this document is to provide analysis of the current services employed at SFA, verify security at network perimeters, and provide emphasis on enabling technologies and solutions which support security within the SFA business model.

- Security Policy and Standards of Configuration - aims at achieving a secure environment by establishing consistency in architecture and to reducing the risk, effect and cost of security incidents. The SFA Security Framework will furnish centralized control to maintain SFA security policy. It will deliver the flexibility to control and manage access through the Security Framework from a central location. These features include an easy to use management interface, configuration of remote sites and monitoring of all systems from a centralized location. Access control rules will be established in accordance with SFA security policy. The SFA Security Framework will provide sophisticated access controls defined through measures such as time, day, user groups, network groups, network interface, inbound & outbound authentication, and encrypted tunnels.

- Security Management / Operations – covers the overall responsibility for the management of the secure enterprise as well as monitoring of the security infrastructure. Within this section, roles and responsibilities will be identified. Central onsite and remote management capability is necessary to accomplish the network administration concerns of SFA. The configuration of remote sites from a centralized location provides an additional layer of administration and control of information security. This is accomplished through use of strong authentication mechanisms and Virtual Private Network (VPN) technology. Analogous to the need for remote administration, the delegated administration of users is essential for efficient systems management.

Management of users inside and outside SFA should be delegated to an infrastructure at the lowest common denominator, such as an academic financial administration group.

- Security Awareness – communicates the security policies and procedures to all employees, business partners and customers to set expectations regarding information security. Awareness programs establish and communicate individual responsibility for protecting the confidentiality, integrity and availability of business assets. The awareness program is used to communicate Information Security policies and standards of configuration to all personnel responsible for handling, administration, or maintenance of systems containing SFA related electronic information.

- Security Compliance - includes all functions necessary to ensure that the security policy and standards of configuration are created, implemented, measured, enforced and updated as required. The SFA Security Framework enables two levels of security compliance. It will monitor the SFA infrastructure for intrusion detection and policy compliance, while the combination of routers and firewalls will verify the authenticity and integrity of Internet users that are attempting contact. The SFA Security Framework will provide fine-grained proxy services that will authenticate, authorize, and control access to limit activity between the two internal and external network interfaces, thus, disallowing any direct communication between the two network interfaces. .

- Security Administration - performs administrative processes, primarily oriented towards managing users of SFA system resources. The SFA Security Framework will have a management interface to allow efficient administration of access rules policy management. Security administrators can set security parameters, control access, and monitor activity through this interface. Access rules allow control of connections based on time, date, user groups, network groups, network interface, inbound & outbound authentication, and encrypted tunnels.

To create a secure domain, all functions provided by the SFA Security Framework must be administered via a common interface. This administrative interface will specify how the requesting user (no matter where located) will be allowed to participate in SFA secure domain. The SFA Security Framework will broker all the underlying network issues and security precautions to make the SFA Extranet, Intranet and Internet secure.

- Security Services for Application Development - supports and enables the development of new security technologies, applications, systems, and business capabilities, with the ability to tie into the Security Framework. The architecture will support Application Security, Authorization, and integration with WebSphere, Applets, Servlets, EJB components, CORBA, Java, and Legacy applications via standards and customizable APIs.

8 Development Architecture

The Development Architecture defines the development tools, methods, standards, and procedures that define the development environment for the ITA. The purpose of the development architecture is to support the tasks involved in the analysis, design, construction, and maintenance of the SFA business applications.

This volume addresses the tools defined as part of the development environment for the ITA. The standards, guidelines, and procedures for using the tools as part of a standard SFA methodology are not included in this volume.

The approach used to define the development architecture for the ITA was based on an analysis of the existing SFA development tools in place at the VDC and ongoing development tasks. In concert with these existing tools the development tools required to develop applications within the ITA environment were identified as well.

8.1. Development Architecture Overview

The Development architecture provides an environment for component-based solutions that support the Analysis, Design, and Construction phases of the development process. It is the combination of development tools, methods, standards, and procedures essential to a comprehensive, integrated environment for developing and maintaining systems. The development architecture provides a starting point for designing and building a development environment, and identifies key concepts and components for the environment.

8.2. Development Framework

The SFA Development Framework is based upon an Integrated Development Environment Architecture (IDEA). IDEA provides a development environment framework and associated guidelines that reduce the effort and costs involved with designing, implementing, and maintaining an integrated development environment.

The development environment is built upon an integrated set of tools and components, each supporting a specific task or set of tasks in the development process. Figure 10 shows the central component, System Building, which is supported by eleven management components.

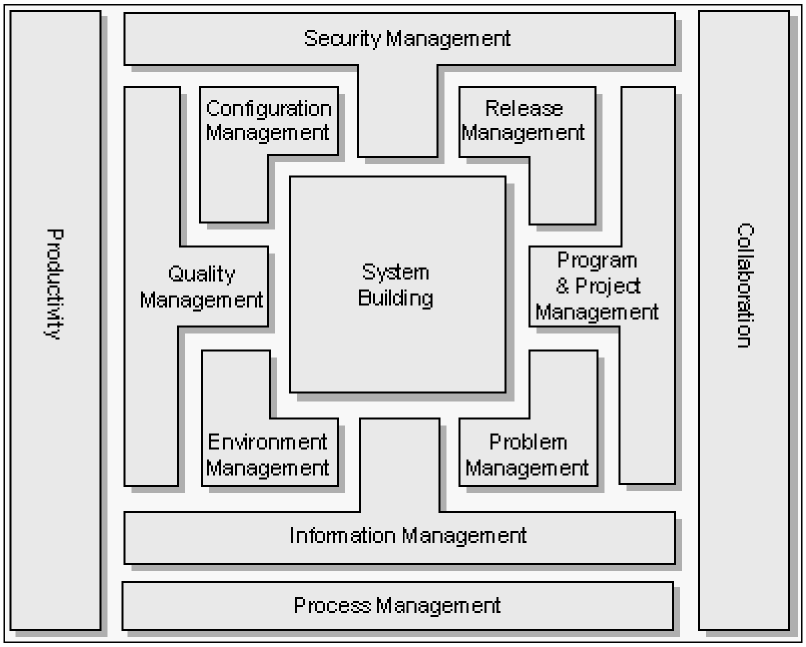


Figure 10 - Integrated Development Environment Architecture (IDEA)

A brief description of the services provided by Development Architecture is listed in the table that follows.

Table 17 – Development Architecture Services

Development Architecture Component	Description
Information Management Tools	Manage the information that supports the entire project – information that is used both in systems building and in other management processes.
Security Management Tools	Enable the development of security components.
Quality Management Tools	Ensure that an agreed-on level of quality in the system is reached. They are also used to provide information and process for improving the quality over time.
Program and Project Management Tools	Assist the management teams in their daily work.

Development Architecture Component	**Description**
Environment Management Tools	Comprised of the following tools to support Environment Management in the development environment. Change Management – Supports the various aspects identifying and managing change in the development environment, the key tool is the Data & Software Distribution which enables automated distribution of data and software to the workstations and servers of the development environment. Service Management – Supports various aspects of supporting and managing the interface with the developers. Service Planning – Planning required to anticipate and implement changes to the other areas: service management, systems management, change management and strategic planning. System Management – Supports the various aspects of supporting and managing the operation of the distributed system
Release Management Tools	Manages the simultaneous development of multiple releases.
Configuration Management Tools	Covers the version control, migration control and change control of system components such as code and its associated documentation.
Problem Management Tools	Pertain to the problem tracking and solution process.
Productivity Tools	Productivity tools provide the basic functionality required to create documents, spreadsheets, and simple graphics or diagrams. Personal Productivity tools are typically packaged as integrated suites of software. These packages provide the basic functionality required to create documents, spreadsheets and simple graphics or diagrams. More recently, the ability to access the Internet and browse electronic documentation has been added to the suite of Personal Productivity tools. - Spreadsheet - Graphics - Word Processor
Collaborative Tools	Enable groups of people to communicate and to share information, helping them work together effectively, regardless of location.
Process Integration Tools	Enforce the correct sequencing of tasks and tools in conformance with a pre-defined methodology.

9 Operations Architecture

The Operations Architecture is a combination of tools and support services required to keep a production system up and running efficiently. Unlike the Execution and Development Architectures, its primary users are system administrators, production support, and service management personnel.

The Operations Architecture volume addresses the tools defined as part of the operations environment for Release 1.0 of the ITA. The standards, guidelines, and procedures for using the tools as part of a standard SFA methodology are not included within this document.

The approach used to define the operations architecture for the ITA was based on an analysis of the existing SFA monitoring tools in place at the VDC and ongoing operations management tasks. In concert with these existing tools, additional tools required to support applications within the ITA environment were identified.

9.1. Operations Architecture Overview

The Operations Architecture is the technology component of the Operations Infrastructure, which is the bundled set of people, process, and technology involved in Operations. As a result, the Operations Architecture framework below provides the tool or technology blueprint for an operations environment.

9.2. Operations Framework

The Operations Architecture Framework contains three main elements listed below.

9.2.1. Component Categories & Components

These are depicted in the middle and upper right element of the framework diagram. The categories shown represent a logical grouping of technology components based on industry drivers or interdependencies of the components. The six component categories are:

- Operations Integration Architecture Components
- Network / Systems Management Components
- Solution Availability Components
- Service Management Components
- Configuration Management Components
- Physical Site Management Components

Each of these categories has associated "Components" within them. Each component provides an operations architecture with specific functionality (e.g. a Service Desk component). Each component has been classified in only one component category even though it may have a tight interrelationship with components in other categories.

9.2.2. Operations Data Architecture

This is depicted on the left side of the framework diagram. This represents where and how operations data is stored, how it is accessed and by whom, where it is used, and how it is distributed.

9.2.3. Operations Execution and Development Architectures

These are depicted along the bottom of the framework diagram. They represent the environments in which operations architecture components are developed, deployed, and operated as shown in the following diagram.

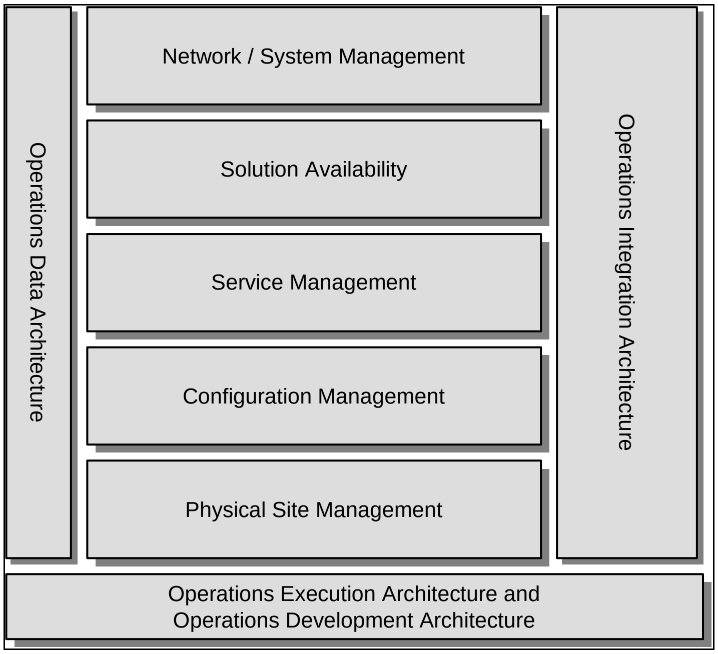

Operations Data Architecture
Network / System Management
Solution Availability
Service Management
Configuration Management
Physical Site Management
Operations Integration Architecture
Operations Execution Architecture and
Operations Development Architecture

Figure 11 - Operations Architecture Framework

A brief description of the services provided by Operations Architecture is listed below:

Network/Systems Management – Includes all components that monitor network and systems performance, process job activity, and diagnose and report failures. The technical architecture hardware and software components addressed by Network / Systems Management includes hardware and software in all Information Technology (IT) environments; Development, Testing, and Execution. In addition the actual IT architecture components include: facilities, hardware, operating system, system software, network, database, custom applications, and packaged applications.

Solution Availability – Ensures availability of IT systems through back-up or redundancy strategies.

Service Management – The Service Management Components include those components that assist the IT organization in providing quality IT service and support. Since Service Management is a process-oriented function it must be kept in mind that these Components address Service Management from only a technical perspective. Includes help desks, capacity planning, user administration, service level management.

Configuration Management – Includes the components which help the IT Operations Environment understand and modify "what is where". These components may either track information about an element configuration and "push or pull" data and software to an element. Information that needs to be tracked includes product licensing information, warranty information, vendor names, logical and physical device information, product configuration tracking, software and version levels, network configuration parameters, physical location, and accounting information.

Physical Site Management -- Ensures that the physical environment is managed and protected against unplanned outages. The Physical Site Management Component Category applies to all environment architectures; Development, Testing, and Execution Components that may be implemented as part of this category include:

- Uninterruptible Power Supply (UPS) / Generator
- Raised Floor
- Fire Suppression & Climate Control
- Wiring / Cabling
- Disaster Recovery

Operations Data Architecture – Represents where and how operations data is stored, how it is accessed, where it is used, and how it is distributed. In many cases parts of the operations data architecture will be predefined by packaged operations architecture components that impose specific logical and / or physical data models. In addition, the operations data architecture is significantly impacted by the operations processes and organization as these dictate who needs access to data and how data will be used. As such, design and implementation of the operations data architecture should always involve teaming with process and organization implementation teams.

Operations Development and Execution Architecture – The Operations Execution Architecture provides the run-time services required for Operations Architecture components to execute. It answers the question "What is the technology environment on which my operations architecture components will run?" Put another way, it is the "infrastructure" for operations architecture components. The Operations Development Architecture provides a unified collection of technology services, tools, techniques and standards for constructing and maintaining components of the operations architecture. It answers the question "What is the technology environment in which my operations architecture components are designed, built and maintained?"

9.3. Operations Architecture Tools Mapping

The following tables map the operations environment tools currently in use or scheduled to be deployed in the SFA operations environment. Each operations architecture component identifies the set of tools within the defined framework.

Table 18 – SFA Operations Environment Tools

Operations Architecture Services	Tools/Comments
Network / Systems Management	Computer Sciences Corporation (CSC) is currently responsible for performing network / systems management at the VDC. The suite of tools that are currently in use will be required to support the network / systems management requirements of Release 1 of the ITA. In addition, the following tools will be used to support system management requirements of specific technical infrastructure applications: • Viador User Administrator Module • International Business Machines (IBM) WebSphere Administrative Console • IBM eNetwork Dispatcher Monitoring Tool • Informatica PowerCenter Server Manager & Repository Manager • MicroStrategy Administrator • IBM MQSI System Administrator • MQSoftware Qpasa!
Solution Availability	CSC currently supports availability of systems located at the VDC
Service Management	TBD--Based upon identification of the organization that will provide Service Management capabilities for the ITA
Configuration Management	• Rational Clear Case • cc:Harvest • Interwoven • Endevor
Physical Site Management	Physical Site Management will not be addressed within the scope of this document. CSC currently handles Physical Site Management for SFA at the VDC. For Release 1.0 of the ITA, the policies, procedures, and processes currently in place at the VDC will be sufficient to support the physical site requirements.
Operations Data Architecture	Oracle and DB2 are the standard Relational Database Management System (RDBMS) for storing operational data.
Operations Development and Execution Architecture	Refer to the Development and Execution Architecture sections of this document to identify runtime services required for the operational components of the technical architecture.

10 Acronyms

Acronym	Description
AFS	Andrew File System
AMI	Application Message Interface
API	Application Programming Interface
CB	Component Broker
CBS	Campus Based Systems
CDS	Central Data System
CFO	Chief Financial Office
CIO	Chief Information Office
CMI	Common Message Interface
CMM	Capability Maturity Model
COE	Common Operating Environment
CORBA	Common Object Request Broker Architecture
COTS	Commercial-Off-the-Shelf
CPS	Central Processing System
DBA	Database Administrator
DLCS	Direct Loan Consolidation System
DLSS	Direct Loan Servicing System
DOE	Department of Education
DWA	Data Warehouse Architecture
EAI	Enterprise Application Integration
EDD	Enterprise Data Domain
EJB	Enterprise Java Bean
EJS	Enterprise Java Server
ETL	Extract, Transform and Load
FFEL	Federal Family Education Loan
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IA	Internet Architecture
IDEA	Integrated Development Environment Architecture

Acronym	Description
IFAP	Information for Financial Aid Professionals
IT	Information Technology
ITA	Integrated Technical Architecture
LAN	Local Area Network
LDAP	Lightweight Directory Access Protocol
LDD	Legacy Data Domain
MDE	Multiple Data Entry
MQ	Message Queuing
NSDLS	National Student Loan Data System
NT	New Technology
ODS	Operational Data Stores
OLAP	On-Line Analytical Processing
OO	Object-Oriented
ORB	Object Request Broker
PDF	Portable Document Format
PEPS	Post-secondary Education Participants System
RDBMS	Relational Database Management System
RFMS	Recipient and Financial Management System
ROLAP	Relational On-Line Analytical Processing
SA	Security Architecture
SAN	Storage Area Network
SFA	Student Financial Assistance
SQL	Structured Query Language
TIVWAN	Title IV Wide Area Network
TO	Task Order
UPS	Uninterruptible Power Supply
URL	Uniform Resource Locator
VDC	Virtual Data Center
WLM	Workload Management